VibHam

An Efficient Program for Predicting Rovibrational Spectra of Diatomic Molecules

Version 1.0

The implementation and development of VibHam was performed by **Gavin A. McCarver** under the advisement of **Robert J. Hinde** at the University of Tennessee, Knoxville

Contents

List of Figures

List of Tables

FAQ - Frequently Asked Questions

How do install VibHam on Linux/MacOS/Windows?

VibHam is available for Linux, Windows, and Mac OS X platforms. To download the necessary python files, please visit our GitHub repository: https://github.com/gmccarve/VibHam. Within our GitHub repository are instructions on how to copy the repository, test the installation of VibHam, and utilize example potential energy and dipole moment curves to get an understanding of how VibHam is run.

I've installed VibHam, how do I run it?

Upon the successful installation of VibHam, the program can be either used interactively with a graphical user interface (GUI) or through a text-based output. To access the GUI, use one of the following commands:

python3 /path/to/VibHam.py -i

python3 /path/to/VibHam.py -Interactive

python3 /path/to/GUI.py

To utilize the text-based output format, a data file must be specified using the *-Data* flag and either the atoms or the masses must be specified using the *-Atoms* or *-Masses* flags. For example, to utilize a potential energy curve of HF found in the data file *HF.txt*, the following command would be used:

python3 /path/to/VibHam.py -Data HF.txt -Atoms H F

The GUI offers a more in-depth examination of the rovibrational properties of diatomic molecules than is afforded by the text-based program. This includes plotting capabilities, the access to data tables, and the ability to view the constructed wavefunctions.

What do I need to run VibHam?

In order to run the VibHam program, all of the python files in the GitHub repository are required as well as an installation of Python 3.7 or newer. Four non-standard python libraries are required to run VibHam: NumPy, pandas, matplotlib, and PyQt5. The latter three are required only for GUI. To install these Python libraries, it is recommended to use the *pip* installation command (see: https://pypi.org/project/pip/).

How do I cite VibHam?

A manuscript for the VibHam program is currently being completed and so following a successful peer review process, we will have a citation available.

How do I make a recommendation or suggestion for VibHam?

Please email Gavin McCarver at gmccarve@vols.utk.edu for recommendations or suggestions.

Can I help contribute to the further development of VibHam?

Please email Gavin McCarver at gmccarve@vols.utk.edu for opportunities to help the further development of VibHam.

What further developments are planned for VibHam?

Two further developments are planned for VibHam: multithreaded processor to make the construction of the Hamiltonian matrices more efficient and moving beyond diatomic molecules to more complex vibrational modes.

1 General Information

1.1 Program Components

VibHam consists of several separate programs that are called as necessary during each run. The following files and modules are includes in the current version of VibHam:

Atoms.py	-	Isotopic mass information
Conversions.py	-	Conversion factors
GUI.py	-	Main program of graphical user interface
Hamil.py	-	Constructs Hamiltonian matrices
Input.py	-	Input program
Interpolate.py	-	Interpolate the energy and dipole curves
Spectra.py	-	Calculate spectroscopic values
VibHam.py	-	Main program for text-based output
Windows.py	-	Program used for external windows in the GUI

2 Calling the Program

The VibHam program is called in the same manner as other Python scripts and codes using the *python* or *python3* command.

Upon the successful installation of VibHam, the program can be either used interactively with a graphical user interface (GUI) or through a text-based output. To access the GUI, use one of the following commands:

python3 /path/to/VibHam.py -i

python3 /path/to/VibHam.py -Interactive

python3 /path/to/GUI.py

To utilize the text-based output format, a data file must be specified using the *-Data* flag and either the atoms or the masses must be specified using the *-Atoms* or *-Masses* flags. For example, to utilize a potential energy curve of HF found in the data file *HF.txt*, the following command would be used:

python3 /path/to/VibHam.py -Data HF.txt -Atoms H F

The GUI offers a more in-depth examination of the rovibrational properties of diatomic molecules than is afforded by the text-based program. This includes plotting capabilities, the access to data tables, and the ability to view the constructed wavefunctions.

3 Input Arguments

The VibHam text-based output allows for fine control of the program using several input arguments or parameters. The only input arguments that are required for each run of VibHam are *-Data* and either *-Atoms* for the identity of the two atomic species or *Masses* for the mass value of each species. The latter allows for the examination of two-body systems that are not strictly diatomic molecules. In this case, some of the output information becomes meaningless such as the spectroscopic constants and Dunham parameters. Each input argument is initiated by adding

'-$*INPUT* $*VALUE*'

after the VibHam.py file is called. Here, $*INPUT* corresponds to the input argument (italicized in the list below) and $*VALUE* corresponds to the value(s) to be assigned to $*INPUT*. Some of the input arguments require one value to be included, some require two values, and some require none. The latter are Boolean style arguments where the presence of the -$*INPUT* argument causes a flag to change from *False* to *True* within the VibHam program.

3.1 List of input arguments

Data - The input datafile to be read in by VibHam. Must be in either two or three columns:

1 - Bond distance values

2 - Energy values

3 - Dipole moment values (optional)

Type	- str
Number of inputs	- 1
Default	- None

R_unit - Unit for bond distance values.

Type	- str
Number of inputs	- 1

Default - "ang"

Options - "ang", "bohr", "m"

E_unit - Unit for energy values.

Type - str

Number of inputs - 1

Default - "hartree"

Options - "hartree", "kj/mol", "kcal/mol", "ev", "j", "cm"

Dip_Unit - Unit for dipole moment values.

Type - str

Number of inputs - 1

Default - "debye"

Options - "debye", "au"

Atoms - Atoms used for the potential energy curve. Up to element 118 is implemented. 2 values required.

Type - str

Number of inputs - 2

Default - None

Masses - Masses of the atoms to be used in the calculation. Must be given in atomic mass units. 2 values required.

Type - float

Number of inputs - 2

Default - None

Isotopes - Isotope values to use. Must match with the dictionary found in Atoms.py. 2 Values required. Must match order of Atoms

Type - int
Number of inputs - 2
Default - None

Energy_Fit - Degree for power series expansion for the energy curve.

Type - int
Number of inputs - 1
Default - 0
Options - 2, 4, 6, 8, 10, 12, 14, 16

Dipole_Fit - Degree for polynomial fit for dipole moment curve.

Type - int
Number of inputs - 1
Default - 0
Options - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11

EigVal - Convergence value for eigenvalues of Hamiltonian Matrix in wavenumbers

Type - float
Number of inputs - 1
Default - 0.1

J - Maximum Value for Rotational Quantum number

Type	- int
Number of inputs	- 1
Default	- 0

Trap - Number of intervals for trapezoid rule for numerical integration

Type	- int
Number of inputs	- 1
Default	- 2000

v - Maximum Value for Vibrational Quantum Number

Type	- int
Number of inputs	- 1
Default	- 20

Print - Level of Printing

1 - Print only converged eigenvalues

2 - Print only converged eigenvalues/vectors.

3 - Print all eigenvalues.

4 - Print all eigenvalues/vectors.

Type	- int
Number of inputes	- 1
Default	- 1
Options	- 1, 2, 3, 4

Charge - Charge of molecule

Type	- int
Number of inputs	- 1
Default	- 0

InterPoints - Number of grid points to include in the polynomial fit. Used for plotting and calculating the minimum of the curve

Type	- int
Number of inputs	- 1
Default	- 100000

Constants - Number of spectroscopic constants to calcualte for a given state.

Type	- int
Number of inputs	- 1
Default	- 3

LoadData - Boolean to try and load in precomputed Hamiltonian Matrices

Type	- bool
Default	- False

Interactive - Boolean to load interactive GUI

Type	- bool

Default - False

i - Boolean to load interactive GUI

Type - bool

Default - False

4 Program Sub-blocks

4.1 Initial Interpolation of Potential Energy Curve

4.2 Power Series Expansion of Potential Energy Curve

4.3 Power Series Expansion of Dipole Moment Curve

4.4 Construction of Hamiltonian Matrices

4.4.1 Harmonic Hamiltonian Matrix

4.4.2 Anharmonic Hamiltonian Matrix

4.4.3 Centrifugal Potential Hamiltonian Matrix

4.4.4 Transition Dipole Moment Hamiltonian Matrix

4.4.5 Stability of Total Matrix

4.4.6 Truncation Error of Finite Hamiltonian Matrices

4.5 Energy Levels and Turning Points

4.6 Dissociation Energy

4.7 Rovibrational Excitations

4.8 Spectroscopic Constants

4.8.1 Vibrational Constants

4.8.2 Rotational Constants

4.8.3 Rovibrational Constants

4.9 Dunham Coefficients

5 Graphical User Interface

6 Some Tips & Tricks

7 Publications Related to VibHam

Bibliography